NASA/TM—2005-213580 ARL–MR–610

New Applications for the Testing and Visualization of Wireless Networks

Robert I. Griffin
RS Information Systems, Inc., Cleveland, Ohio

Michael A. Cauley
Glenn Research Center, Cleveland, Ohio

David P. Pleva
Verizon Federal Network Systems, Arlington, Virginia

Marc A. Seibert
Glenn Research Center, Cleveland, Ohio

Isaac Lopez
U.S. Army Research Laboratory, Glenn Research Center, Cleveland, Ohio

March 2005

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.
- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.
- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at ***http://www.sti.nasa.gov***
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA Access Help Desk at 301–621–0134
- Telephone the NASA Access Help Desk at 301–621–0390
- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076

NASA/TM—2005-213580 ARL–MR–610

New Applications for the Testing and Visualization of Wireless Networks

Robert I. Griffin
RS Information Systems, Inc., Cleveland, Ohio

Michael A. Cauley
Glenn Research Center, Cleveland, Ohio

David P. Pleva
Verizon Federal Network Systems, Arlington, Virginia

Marc A. Seibert
Glenn Research Center, Cleveland, Ohio

Isaac Lopez
U.S. Army Research Laboratory, Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center

March 2005

Acknowledgments

This research was conducted under NASA's CICT/CNIS Information Power Grid Task. The author would like to express his gratitude to Isaac Lopez and Greg Follen. Alex Littel and Collin Corlett of Intellidyne provided us with access to the Vivato panel antenna and assisted in its testing. Finally, the author would like to thank Paul Catalano and Scott Townsend for the donation of various and sundry sage advice.

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100

Available electronically at http://gltrs.grc.nasa.gov

New Applications for the Testing and Visualization of Wireless Networks

Robert I. Griffin
RS Information Systems, Inc.
Cleveland, Ohio 44135

Michael A. Cauley
National Aeronautics and Space Administration
Glenn Research Center
Cleveland Ohio 44135

David P. Pleva
Verizon Federal Network Systems
Arlington, Virginia 22209

Marc A. Seibert
National Aeronautics and Space Administration
Glenn Research Center
Cleveland Ohio 44135

Isaac Lopez
U.S. Army Research Laboratory
Glenn Research Center
Cleveland Ohio 44135

Abstract

Traditional techniques for examining wireless networks use physical link characteristics such as Signal-to-Noise (SNR) ratios to assess the performance of wireless networks. Such measurements may not be reliable indicators of available bandwidth. This work describes two new software applications developed at NASA Glenn Research Center for the investigation of wireless networks.

GPSIPerf combines measurements of Transmission Control Protocol (TCP) throughput with Global Positioning System (GPS) coordinates to give users a map of wireless bandwidth for outdoor environments where a wireless infrastructure has been deployed. GPSIPerfView combines the data provided by GPSIPerf with high-resolution digital elevation maps (DEM) to help users visualize and assess the impact of elevation features on wireless networks in a given sample area.

These applications were used to examine TCP throughput in several wireless network configurations at desert field sites near Hanksville, Utah during May of 2004. Use of GPSIPerf and GPSIPerfView provides a geographically referenced picture of the extent and deterioration of TCP throughput in tested wireless network configurations. GPSIPerf results from field-testing in Utah suggest that it can be useful in assessing other wireless network architectures, and may be useful to future human-robotic exploration missions.

Recommendations

- GPSIPerf should be used to characterize the TCP throughput in wireless network architectures at field research sites.
- Visualization of wireless network characteristics should be done in terms of its geographic context.
- More work needs to be done to create and incorporate additional indicators of wireless network speed, signal strength and associative capabilities in GPSIPerf.
- Concerted effort should be expended to create and use radio wave propagation models that are accurate over distances (<1 km) appropriate to wireless networks.

1. Background

Building and maintaining wireless networks is not an easy task. Two separate approaches may be used together to best determine the configuration of hardware in a sparse, outdoor wireless network: modeling and empirical determinations.

Modeling wireless networks is difficult and any given model's best estimate of the signal strength and the quality of such networks is often inaccurate. Engineers of wireless networks, lacking models sufficient to their needs, are often relegated to empirical determinations of these network characteristics. Such determinations utilize varying arrays of hardware and software and may be chosen over wireless

network modeling when planning and maintaining wireless networks. Some of these hardware and/or software packages are quite expensive while others are relatively inexpensive, but may not adequately characterize the wireless network in question.

For example, these tools may enable autonomous robotic planetary surface explorers to map an exploration terrain before or during future NASA human-robotic missions. This information is useful because human explorers require constant communication during extra-vehicular activity (EVA). GPSIPerf and GPSIPerfView may allow mission control and surface explorers to predict, identify, and measure gaps in communications availability, understand their actual exploration limits spatially, or understand where modifications to exploration communications infrastructure are required to complete a particular objective.

1.1 Existing Software Solutions

The number of wireless networking software utilities that are currently under active development serves is a good indicator of the amount of attention that field of wireless networking is currently receiving. Most of these applications perform various types of network auditing utility.

NetStumbler (ref. 1) and Kismet (ref. 2) are examples of two popular wireless network-auditing utilities. Ostensibly, these tools may be used to:

- Locate wireless network AP's (access points)
- Verify/identify network configuration.
- Identify areas with strong and weak wireless radiofrequency (RF) signals from the access point.
- Detect interfering networks.
- Assist with pointing directional antennas in complex network architectures.

These applications are capable of reporting wireless network parameters such as MAC Address, Service Set ID (SSID), Name, Channel, Speed, Encryption Status and Signal-to-Noise ratio. Kismet, unlike NetStumbler is a passive monitor; in addition to identifying access points without announcing itself, it can also detect hidden networks on the basis of analyzing intercepted packets.

Few, if any wireless networking utilities attempt to measure bandwidth at the application level. Most products rely on layer-1 RF signal strength as the link quality measurement. Signal strength is an instantaneous measurement that can vary dramatically from one second to the next. Throughput, in terms of bandwidth, is an expression of the total data transport capability of wireless networks over a period of time. It is approximated by parameters such as "Speed" that are reported by NetStumbler.

The term "goodput" narrows the definition of throughput to be *the empirical amount of data actually usable by applications over time*. Goodput is typically some fraction of the total throughput. Dropped frames, retransmission of data, and the addition of protocol headers in the IP stack are all elements that reduce the usable throughput over a network link. Bit error is especially common over wireless links and typically results in a loss of TCP throughput. Wireless network engineers may be interested in throughput, but software developers and operators should be more interested in the goodput of a network.

The National Laboratory for Applied Network Research (NLANR) Distributed Application Support Team (DAST) has created a utility, iperf, which measures the maximum TCP bandwidth of any existing network connection (ref. 3). Measurement of TCP bandwidth using iperf is an indicator of goodput. Unfortunately, iperf, by itself, does not include or make simple the inclusion of Global Positioning System (GPS) data. The use of GPS data in other wireless utilities such as NetStumbler and Kismet aids in establishing the geographical extent of wireless networks with GPS reference data.

The experiments performed in this report made use of wireless equipment based on the IEEE 802.11b standard. The GPSIperf tool measures "goodput" achieved at the TCP layer, so this tool is independent of the layer-1 (PHY) wireless standard employed. This means GPSIPerf could also be used to measure the performance of large open area Internet Protocol-based wireless technologies, including 802.16 (WiMAX), 802.20 (MBWA), and so on. GPSIperf only requires that the client nodes be receiving adequate GPS signals, be within a wireless coverage area, have a GPSIPerf server running somewhere on the network, and that the client maintain an active TCP connection to the GPSIPerf server.

The development and use of networked wireless applications could benefit by measures of goodput combined with GPS data. These combined measurements can help to answer questions that the increasing use of wireless networks has brought to our attention such as:

- Where will my application work?
- What kind of data throughput can I expect?
- How long will this download/upload take?

The responses to these questions are more germane to the developers and operators of applications than are the speeds reported by access-points or measurements of signal-to-noise ratio provided by other wireless utilities. These indicators are primarily hardware level estimates of wireless service quality. What is required is software driven estimates similar to those provided by "iperf."

1.2 New Tools

This work describes two new applications for the testing and visualization of wireless network performance. These tools, when used in tandem, may be used to plan, deploy and test the efficiency of wireless networks in field research settings.

GPSIPerf was developed to pair Global Positioning System (GPS) data with measurements of wireless network performance. Specifically, GPSIPerf measures network maximum TCP bandwidth. GPSIPerf gathers bandwidth measurements *via* the NLANR Distributed Application Support Team's "iperf" utility. GPSIPerfView is a tool that was developed to visualize GPSIPerf measurements in a geographical context. United States Geological Survey (USGS) (ref. 4) Digital Elevation Map (DEM) data is used to create a virtual terrain upon which GPSIPerf data may be mapped.

GPSIPerfView also includes a configurable tool that projects the results of wireless network signal loss calculations onto a Digital Elevation Map (DEM). Calculations of signal loss are derived from the National Telecommunications and Information Administration's (NTIA) Irregular Terrain Model (ITM) for radio propagation (ref. 5).

2. GPSIPerf Implementation and Testing

Development and testing of GPSIPerf and GPSIPerfView took place in several different locations. Much of GPSIPerf was written during the Fall of 2003 and Spring of 2004. However, some of its development occurred *in situ* during GPSIPerf testing. This paper will address its capabilities and the approaches used for the implementation of specific mechanisms within GPSIPerf rather than detailing a history of its development.

Development of GPSIPerfView occurred after the spring of 2004 field-testing during the late spring and summer of 2004. GPSIPerfView provides an interface by which to view previously gathered GPSIPerf results although it has not been thoroughly tested under field conditions.

Both GPSIPerf and GPSIPerfView were developed using Microsoft Visual Studio .Net 2003. Operability of these applications was tested on Windows 2000 and Windows XP operating systems.

Hardware used in the preliminary development and testing of GPSIPerf is as follows:

- DeLorme USB Earthmate 2 GPS Receiver (ref. 6).
- Sager Notebook Model 8887 (ref. 7).

Wireless network connectivity on the laptop was established using a PRISM-based chipset with Intersil (version 1.07.37) network drivers.

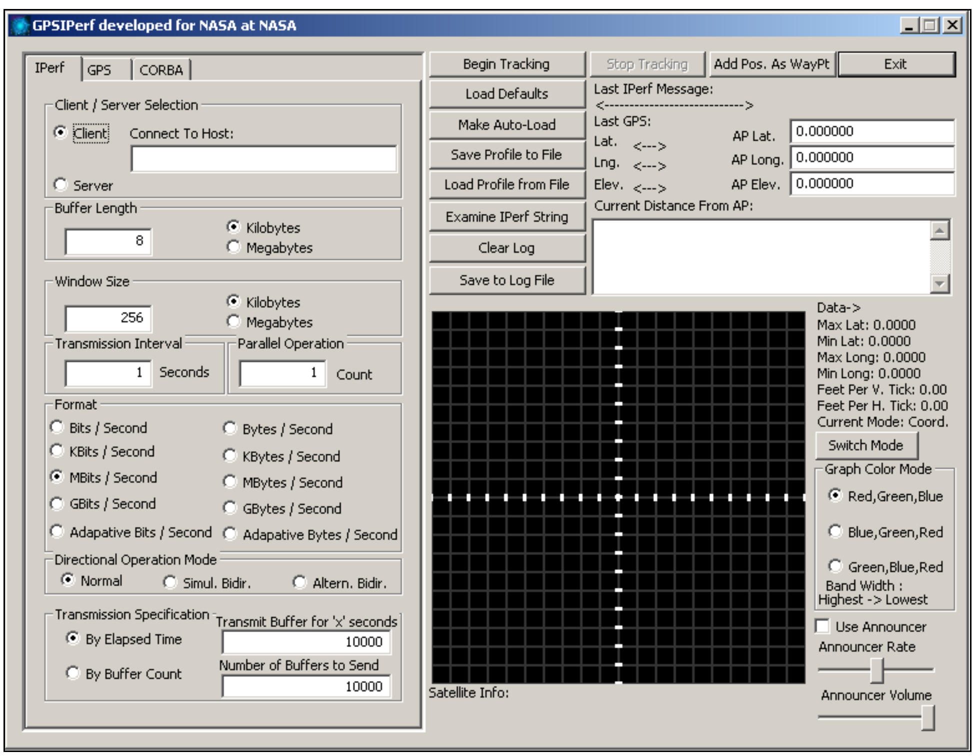

Figure 1.—The GPSIPerf Interface—The GPSIPerf interface offers users the ability to configure iperf, GPS data acquisition and CORBA communications. A simple X-Y plot is used to visualize GPSIPerf measurements over time and over geographic location (geographic location shown above). Users may also have GPSIPerf announce relative bandwidth percentages aloud when CRT/LCD visibility is limited (e.g., in bright daylight). Logs and specific configurations may also be loaded by users.

2.1 GPSIPerf

This section introduces and describes the first of the software components used for wireless throughput investigations. Descriptions of field testing can be found in section 3.

2.1.1 GPSIPerf—User Interface.—GPSIPerf provides an interface to configure GPS data acquisition and iperf parameters as well as a simple visualization and logging tool for correlating GPS location with TCP throughput measurements. GPSIPerf's key features are enumerated here and depicted in figure 1.

Users are given the ability to configure and execute iperf clients. These clients interact with running iperf servers located at the opposite end of a wireless link. Cross-checks of the iperf configuration against command line parameters are available for users who are familiar with the simple command line executable provided by NLANR.

Users may track the bandwidth and location from a configured interface by merely clicking the "Begin Tracking" button. While actively tracking, a properly configured GPSIPerf process is able to record location and iperf bandwidth measurements. These measurements are updated every second in logs, text and visually on a scatter plot. Furthermore, users may choose to mark points of interest during tracking mode using the "Add Pos. as WayPt." Button. Way Points are displayed visually on the positional scatter plot and can be stored in separate log files.

Simple means are made to store and auto-load configuration sets. File-based storage of configurations must be explicitly selected before they are loaded, whereas auto-loaded settings are stored in the registry and are used to populate the various parameter values when GPSIPerf is first executed.

GPSIPerf session log files may be saved either in a binary format or as comma-separated volumes (CSV). CSV files may be easily loaded into Excel and other data processing programs for post-collection analyses.

Recently, the ability to establish a connection with a remote CORBA-based location/bandwidth service has been added to the GPSIPerf interface. GPSIPerf-based reports of bandwidth and position from multiple GPSIPerf clients can be logged and/or tracked from the CORBA server. Additionally, our visualization software, GPSIPerfView, can contact the server to determine the whereabouts and most recent throughput of all GPSIPerf clients.

2.1.2 GPSIPerf—GPS Data Acquisition.—GPS libraries were used that were capable of parsing NMEA (National Marine Electronics Association) formatted output from a GPS Receiver connected to a RS–232 (COM) communication port.

It was determined early on in the software development process that COM port access to GPS data was highly desirable as software solutions implementing such access would be able to utilize a wide array of GPS receivers. Unfortunately, there were few choices for inexpensive, self-powering (i.e., no batteries required), light-weight GPS receivers. The DeLorme USB Earthmate 2 offered the best solution in terms of energy-efficiency, size, weight and expense.

The choice of the DeLorme GPS receiver added some complications in terms of data access due to its use of USB ports as opposed to the more common COM ports. It was necessary to install DeLorme's Earthmate Drivers (ref. 8) using an option to add support for 2nd Party applications for development and testing with this unit. This installation option created a virtual COM port through which GPS Data could be transferred to a non-Delorme Application.

Data access to this virtual COM port is somewhat problematic as the current implementation of the COM port access code in GPSIPerf does not seem to initialize the GPS unit properly. Java-code utilizing the Java Serial Communications Application Programming Interface (API) is used to perform a one-time initialization of the GPS unit at 4800 baud 8–N–1. NMEA 1.8.3 strings were reliably produced by the DeLorme GPS unit following this initialization step. GPSIPerf was capable of creating connections to the COM Port with parameters specified in the user interface once successful initialization had occurred (fig. 2). The default parameters provided were usually enough to get the GPS unit up and running.

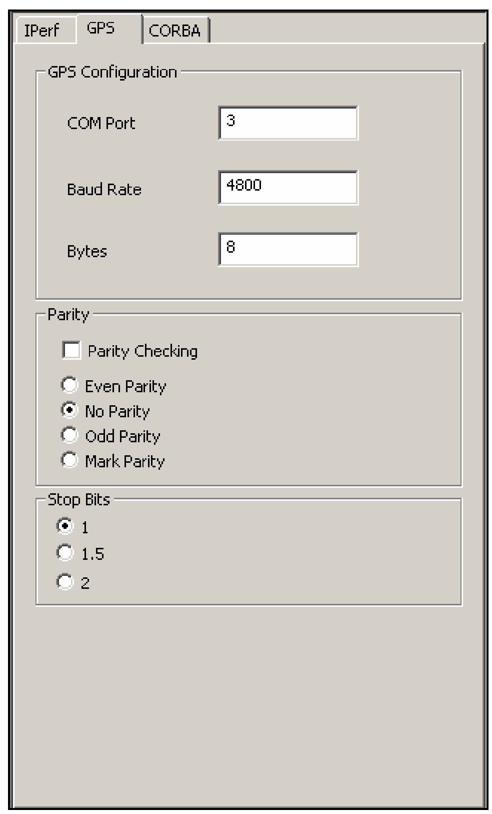

Figure 2.—The GPSIPerf GPS Configuration Interface—Relatively little information must be entered in this tab to properly configure the GPS for data acquisition.

2.1.3 GPSIPerf—IPerf Modifications.—The TCP Bandwidth network performance monitor, iperf, was used to make measurements of TCP throughput. This program was compiled from source code provided by NLANR/DAST web site and executed in a separate thread from that of the GPSIPerf user interface. The iperf executable was started using command line parameters derived from options selected through the user interface (fig. 3). Output from the iperf was logged using output redirection classes. Finally, a mechanism for cleanly shutting down the iperf client using file semaphores was added in order to avoid segmentation faults and premature termination of the iperf server.

Configuration of GPSIPerf to recreate a usable string of the command-line parameters was trivial and did not require modifications to iperf source code. The most complicated task in the execution of iperf was properly redirecting its output to the GPSIPerf executable for collation with GPS data and reporting. This redirection was handled within the GPSIPerf code rather than that of iperf.

Implementations of file-semaphore shutdown mechanisms as well as the new logging output formats did require some small modifications of the iperf code. Changes to the output procedure were minimal. The first of these changes reduced the header reporting to once per execution of iperf as opposed to the reporting of header information every 20 lines. The second change to the output procedure modified the TCP output string to contain the word "data" at the beginning of every reporting line. This word was recognized by the GPSIPerf output parser as indicating that the data coming from the iperf could be used in reporting the Bandwidth measurements.

Changes to iperf that provided a mechanism for clean shutdown of the iperf client as dictated by the user's interaction with the GPSIPerf executable were a little more complex. A separate thread (Quitter) was created that polled for the existence of a file named "quitter.quit" in the execution directory. This empty file was created whenever the user prompted GPSIPerf to "Stop Tracking" or when GPSIPerf was terminated during active tracking. When that file existed this thread would alert all other threads to stop and signal iperf shutdown. This implementation allowed for shutdown of the iperf client in a manner that avoided iperf server segmentation faults and premature termination. Earlier, elegant solutions involving piped signals and drastic measures such as external process termination had both failed to produce the desired results on the iperf server-side.

Overall the configuration of and alterations to iperf proved to be very easy. Readable and well-maintained code made changes easy and the flexibility and power of the initial code ensured that no new options for reporting need be established.

2.1.3 GPSIPerf—CORBA Integration.—Users of GPSIPerf may wish to report their bandwidth and location to a central service for potential use by other applications. The integration of the Common Object Request Broker Architecture (CORBA) into GPSIPerf allows users to send small amounts of data to such a service.

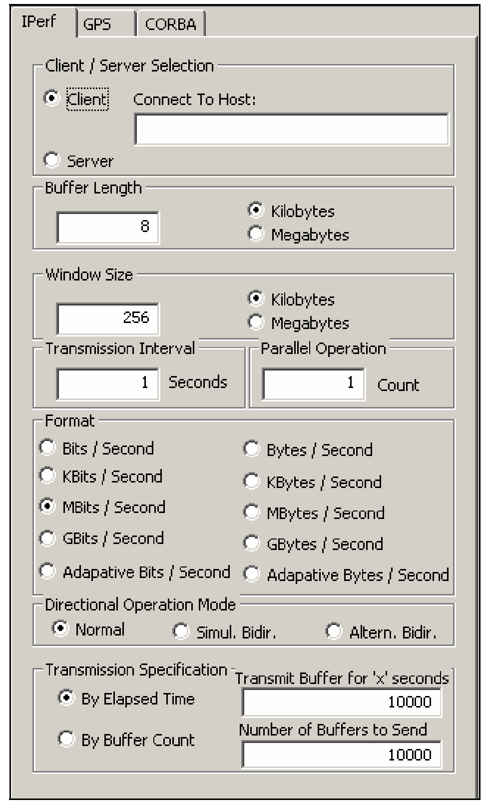

Figure 3.—The GPSIPerf iperf Configuration Interface—The most common IPerf configuration options are available in this tab to configure iperf for data acquisition.

The CORBA is a middleware specification for creating distributed computing frameworks created by the Object Management Group (ref. 9). Mico (ref. 10) is an open source implementation of the CORBA specifications that has proved useful in our experience (ref. 11). Mico version 2.3.11 was used as the backbone CORBA implementation for GPSIPerf and GPSIPerf Data server inter-process communications.

The interoperability of computing components using CORBA is defined by an IDL (Interface Definition Language) document. This document serves as a contract between client and servers. The IDL document used for GPSIPerf services is relatively simple (listing 1). Typical reporting of GPSIPerf client to a GPSIPerf Data Server consists of a single "AddPoint" call. Bandwidth used for CORBA messages is not figured into the final TCP bandwidth provided by the iperf executable. However, the size of the data payload of this message is less than 200 bytes and is transmitted by GPSIPerf once every second when updated positional and bandwidth data are available (i.e., during active tracking). The impact of these messages on throughput estimates is, at this time, thought to be negligible.

The configuration of GPSIPerf to connect to a data server was relatively simple. Users needed to specify only the desire

```
interface GPSIPerf_Data
{
      struct GI_Point {
             long Time;
             double X;
             double Y;
             double Az;
             double BW;
             long NumSats;
             long Fix;
             double PDOP;
             double VDOP;
             double HDOP;
             string UTMZone;
             long DS;
      };
      typedef sequence <GI_Point> GI_List;
      typedef sequence <string> GI_Clients;
      void AddPoint(in string client, in GI_Point data);
      void AddData(in string client, in GI_List data);
      void GetData(in string client, out GI_List data);
      void GetPoint(in string client, out GI_Point data);
      void GetClients(out GI_Clients clients);
};
```

Listing 1.—Interface Definition Language document used to pass GPSIPerf data to a central service for later use by other clients such as GPSIPerfView. The highlighted "AddPoint" call is used by GPSIPerf to register itself and add data to a dataset maintained by a remote GPSIPerf Data Server. "GetPoint" and "GetClients" are used by GPSIPerfView to update client positions.

to use CORBA, and the hostname and port of the GPSIPerf Data Server under a "CORBA" tab that was similar to those of "IPerf" and "GPS" configuration tabs. Successful CORBA connections allow this configuration data to be stored and used as "Auto-Load" defaults for future use.

Data was not sent to the server until Clients began to track. Tracking initialized the connection to server. Once the connection to the server was successfully established data was on each periodic update of the user interface (1 second intervals).

Implementation of the CORBA client interface was relatively simple under Windows as the interface data structure definition (i.e., GI_Point) closely followed a structure that was used often within the GPSIPerf code. Integration of CORBA into the GPSIPerf code was further facilitated by the Windows Microsoft Foundation Classes API. Using this API dispensed with the need for threading requirements that the periodic calls to the server might have caused. Windows "SetTimer" and "KillTimer" calls were used to mimic periodic calls to "AddData" that might otherwise have been performed by a concurrent thread. It is worth noting that, during the system timer events GPSIPerf visualization, log and textual data were updated, in addition to adding data to the GPSIPerf Data server. Although the granularity of the Windows timers can be somewhat coarse (milliseconds), it was deemed sufficient for our use.

2.2 GPSIPerfView

The goal in the development of GPSIPerfView was to provide users with a means by which GPSIPerf data could be regarded in terms of the geographic setting within which the data was derived. United States Geological Survey (USGS) Digital Elevation Maps (DEM's) were used to provide a geological context for field observations.

The GPSIPerfView interface offers users the following capabilities:

- DEM loading and visualization (fig. 4).
- GPSIPerf CSV log loading and visualization.
- Irregular Terrain Model Radio Loss calculations and visualization.

Additionally, GPSIPerf has the ability to create CORBA connections to get "live" GPSIPerf client information from a GPSIPerf Data Server as well as utilities for object control, DEM subset selection, application of shaders/textures to DEM data, marker selection, and the extraction of application screen shots. This set of functionality helps to make the GPSIPerfView application a powerful tool for the investigation, planning, maintenance and testing of wireless networks to be created and used in field research.

2.2.1 GPSIPerfView—DEM Data.—USGS DEM data is readily available from a number of commercial and free sources. Much of this data has been converted to the ISO 8211 Spatial Data Transfer Standard (SDTS) (ref. 12). SDTS data is arranged in a set of files that are compact and well-organized. However, deriving DEM data from these file is not a trivial task. Two different freely available libraries offer solutions to SDTS data access. The first of these libraries SDTS++ is implemented in C++ (ref. 13). This library relies heavily on stream capabilities provided to it by underlying Boost libraries (ref. 14). The second library, fips123, is an older SDTS library developed in C (ref. 15). It use has been deprecated in favor of the SDTS++ libraries.

During our implementation of SDTS data access it was found that fips123 provided faster load times and was more stable than the SDTS++. On average fips123 library

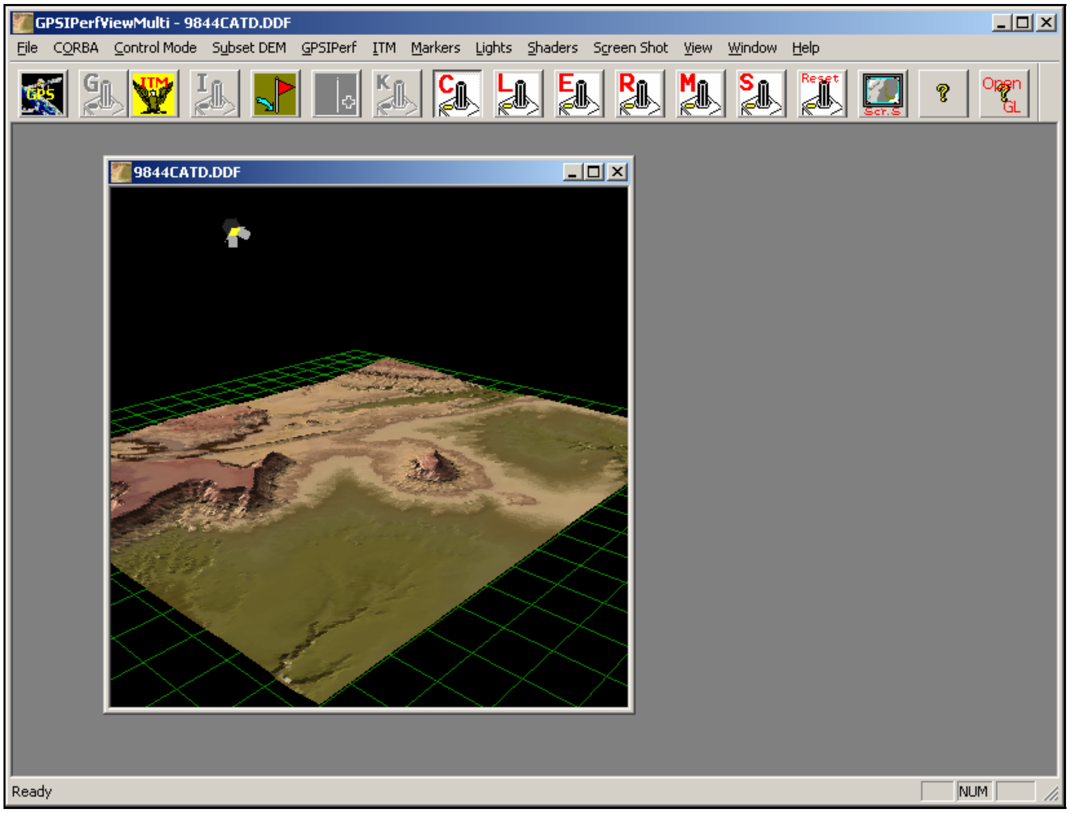

Figure 4.—GPSIPerfView's Interface. GPSIPerfView supports opening and manipulation of the data from multiple DEM data files and GPSIPerf data logs. Individual views of DEM's like that of Factory Butte, Utah (shown above) can be shaded or textured according to a number of different schemes. The coloration scheme for this DEM was derived from a loaded image file.

implementations of DEM data loading methods performed 4 to 5 times faster and loaded the range of DEM's that we used for testing more reliably. In rare cases, some SDTS files caused SDTS++ library-based loading implementations to produce page faults. The cause of these faults is unknown but was reported to SDTS++ mailing lists and remains unremedied at the time of this writing.

The underlying visualization framework for DEM data was used by all other visual components in GPSIPerf. This visualization technique can be separated into two distinct parts: the rendering context or view and the scene graph used to render DEM and other data to the screen. OpenGL was chosen as the underlying rendering library for DEM data as there was plenty of support available for OpenGL development both from external (e.g., internet) and in-house sources (ref. 16). OpenGL visualization in GPSIPerfView's interface used the GLEnabledView code originally created by Alessandro Falappa (ref. 17). This class provided a well-documented entry point in to the Window's OpenGL rendering context and enabled rapid development of mouse and keyboard control procedures.

Rendering of DEM scene-graphs within individual views (fig. 4) and the two-dimensional dialog rendering of DEM data (fig. 5) are heavily derived from code found in Jamie Moyer's Linux-based DEM visualization application: *kdem* (ref. 18). This code was ported to a Windows-based library; GraphDEM, and supported rapid development of many new visualization items and controllers. Support for views and controllers such as those GPSIPerf data, Irregular Terrain Model results and map markers was primarily provided as subclasses derived from this framework of base classes.

2.2.2 GPSIPerfView—GPSIPerf Data.—The goal of viewing GPSIPerf data in the appropriate geographical context was achieved through decomposition into three subtasks:

- Loading of the GPSIPerf Data.
- Referencing loaded positional information into the appropriate DEM areas.
- Visualization and control of GPSIPerf Data on the surface of the Digital Elevation map.

Loading of GPSIPerf log data was accomplished with little difficulty. Writing routines for loading the comma-separated fields in the files into a structure similar to the GPSIPerf Data Server's IDL GI_Point data was very straightforward. However, users had to be given the ability to load multiple log files at once. This required an additional identifier in the structure of each data point to indicate the data's source.

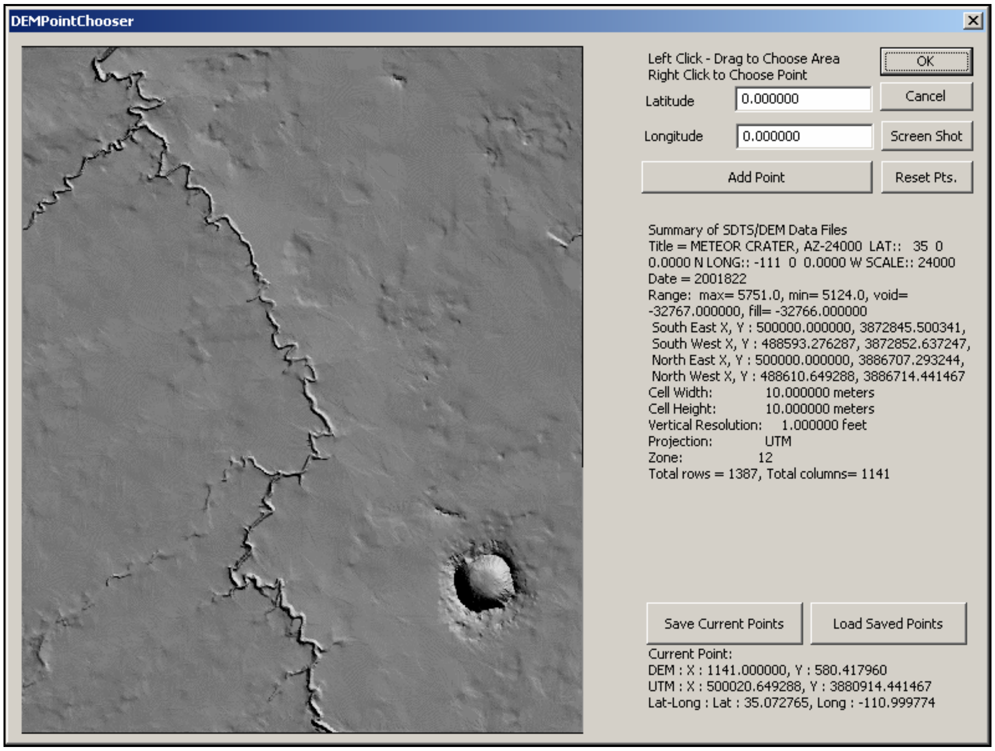

Figure 5.—Two-dimensional rendering of a loaded DEM. This interface provides users with the ability to choose subsets using a mouse click and drag technique as well as select key latitude and longitude points with the mouse or by keyboard entry. The interface is versatile and aids users with Irregular Terrain Model Beacon, Point Marker and Subset selections.

Loaded GPSIPerf data was next normalized to the extents and granularity of the DEM. The typical resolutions of DEM data that were used was 10 or 30 m. GPSIPerfView provided a separate means (i.e., a resolution controller) through which a user could decrease or increase the resolution of the rendered DEM. Rendered resolution could not be increased above that of the original SDTS data, but could be dropped to a single representative point. This complicated the referencing of positional data to the grid, but allowed for decreased rendering times for very large, high resolution DEM’s. GPSIPerf Data points were aligned according to the rendered DEM resolution rather than the absolute resolution to maintain positional integrity. Finally, GPSIPerf data for locales not encompassed by the bounds of the DEM to which they were assigned were ignored.

Visualization of GPSIPerf data used some unique approaches (fig. 6). The first of these was a user-selected color scheme. Red, green, blue was the default color scheme for indication of the relative TCP throughput of the data. Red spheres by default represented the greatest relative throughput, whereas blue spheres indicated the lowest relative throughputs. Users could rearrange the order of the RGB components of this scheme or opt to use black/white as the highest and white/black as the lowest bandwidths and shades of grey for those lying in between. Sets of data that were loaded from different log files were bounded by distinctly colored dashed boxes to aid in their differentiation.

Another unique visualization capability was provided in allowing the user to change the size, and thus visibility of the spheres. Finally, users were able to lift GPSIPerf data *en masse* from the DEM data using a simple mouse movement or keyboard control.

2.2.3 GPSIPerfView—The Longley-Rice Irregular Terrain Model.—The Longley-Rice Irregular Terrain Model (ITM) (ref. 19) was used in GPSIPerfView to provide a qualitative context in which users could gauge apparent increases and decreases in GPSIPerf TCP throughput. ITM is generally considered an industry standard for prediction of the median attenuation of a radio signal as a function of distance and the variability of that signal in time and space. This attenuation is reported in terms of decibel (dB) loss. However, ITM solutions do not incorporate losses due to Fresnel zones or multipathing and are limited to distances between 1 and 20 km.

The ITM model is configured using a number of different user-specified parameters. The values of these parameters were selected through a dialog which was shown when the user chose to run an ITM Model (fig. 7). The user was also provided with the opportunity to save and load other configurations in addition to hand entering the profiles.

The calculations used for point-to-point radio-wave attenuation solutions were derived from C++ code developed

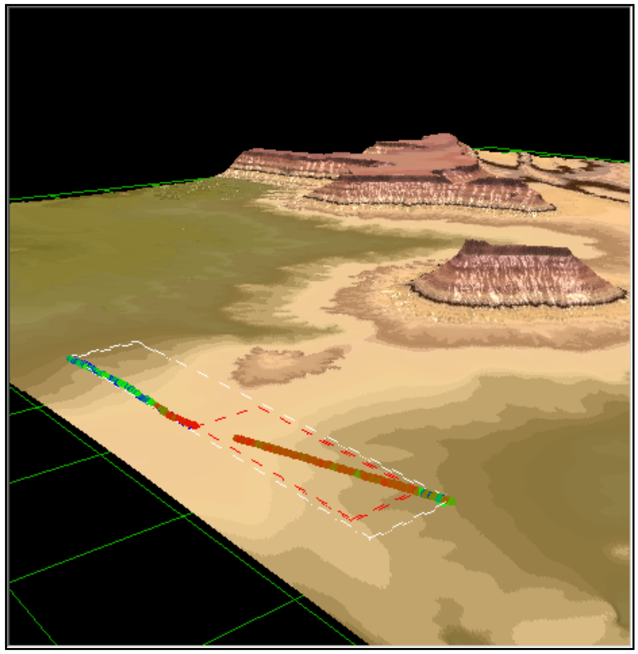

Figure 6.—GPSIPerf data successfully combined with USGS DEM data. This figure shows data from three separate Tropos, Inc., hardware testing runs of GPSIPerf in Factory Butte, Utah. Following the WiFi "hot spot" notion, green (cool) and blue (cold) spheres indicate areas of increasingly low TCP throughput relative to those of the high throughput red (hot) spheres. Differentiation of each of these data series is aided by providing different colored bounding boxes for each of the datasets.

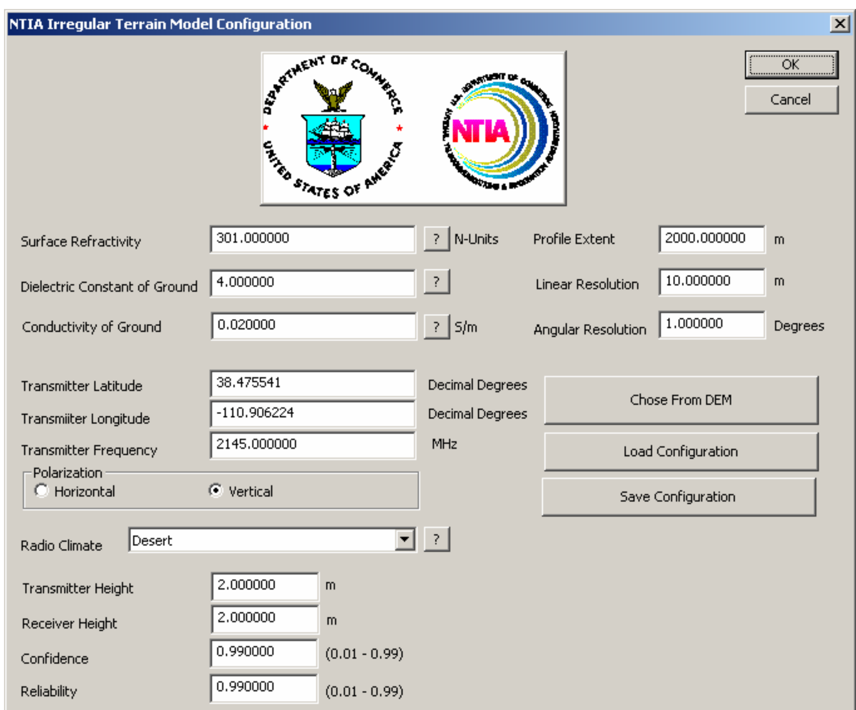

Figure 7.—Configuration Dialog for the Irregular Terrain Model's calculation of radio signal attenuation. Users can choose the question mark boxes for information and suggestion on values for Surface Refractivity, Dielectric Constant, Conductivity and Radio Climate. Latitude and longitude may be selected from a two-dimensional depiction of the DEM (fig. 5). The horizontal extents, radial resolutions and linear resolutions of the solutions are also selected here.

by Fred Najmy and Alaka Paul at the National Telecommunications and Information Administration/Institute for Telecommunication Services (NTIA/ITS) (ref. 19). Only two changes were made to the code prior to its inclusion in GPSIPerfView. First, the model, originally supplied as code for a dynamic link library was converted to a static library. Second, the model was modified to return a qualitative indicator of the solutions (i.e., line of sight, troposcatter dominant, etc.)

ITM solutions are performed iteratively. Vertical profiles were first derived from the DEM. The chosen transmitter position served as one endpoint of the profile. Other elevation points in the profile were sampled at increasing distance radiating from the center point at specified intervals over a specified range. The increment in distance and maximum range were selected by the user as Profile Extent (m), and Linear Resolution (m), respectively. The process of creating elevation profiles was repeated over 360° at an interval selected by the user as Angular Resolution. Points in the elevation profiles that did not correspond exactly to DEM data samples were calculated using interpolation from surrounding, known DEM elevation points.

At each point of each elevation profile a solution for radio-wave attenuation was calculated. Thus, for a horizontal extent of 1000 m with a linear resolution of 10 m and an angular resolution of 1 degree, 36000 (360 · 100) calculations were performed. The results and profiles used for all calculations are written to a single file for later review upon successful completion of the ITM iterations.

Solutions for the model at horizontal distances less than 1 km are not strictly valid. However, qualitative comparison with solutions at positions at or beyond this inner limit indicates strong coherence with solutions located within this limit. Qualitative indicators such as line-of-sight versus single or double horizon troposcatter or diffraction dominance as well as the calculated decibel loss are consistent with the line of sight indicators and increasing linear distance from the transmitter's origin.

The results of these calculations were added to the DEM scene for rendering. Qualitative solution polygons were placed at the horizontal extent of each radial arm. These color and shape of these polygons indicated line of sight, double or single horizon and troposcatter or diffraction dominance. Additionally, solutions for each point are used to indicate relative levels of signal loss. Points with the lowest signal loss (or best transmission levels) are red whereas points with higher loss progress through green to blue (fig. 8).

A separate dialog was developed to provide the user with another visualization solution for viewing individual ITM result profiles. This dialog allowed the user to view precise heading and elevation profiles for each of the profiles in the ITM solution. Additionally, the user has the ability to take screen shots of individual profiles for later examination (fig. 9).

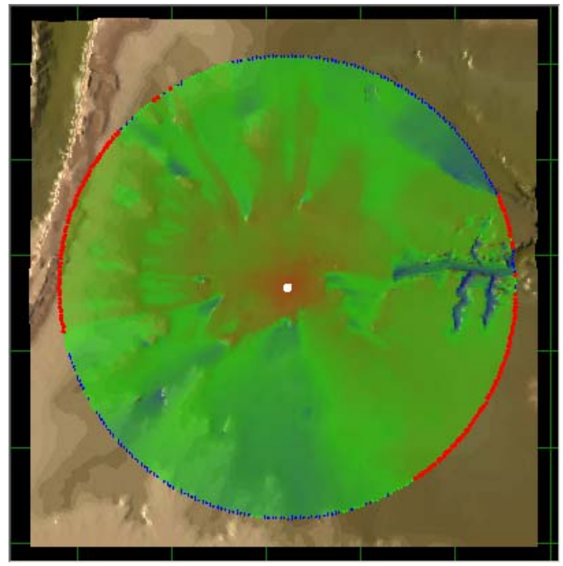

Figure 8.—Results of ITM calculations using DEM data from Factory Butte, Utah. Vertical elevations were sampled over a 2 km radius at 10 m linear resolution once per degree over 360° (72000 calculations) were used to create this image. Red areas nearest the transmitter center point indicate good radio-wave propagation. Lower transmissions indicated by the blue areas in the canyon reflect the loss of line of sight on radio propagation.

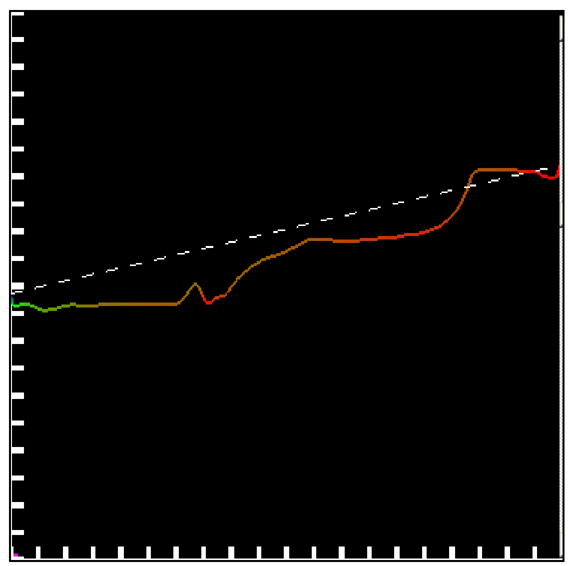

Figure 9.—Elevation profile from the ITM Results dialog. The radio transmitter is located on the left vertical axis. The white dashed line is the line of sight from the transmitter to the receiver. Red areas on the profile indicate areas of relatively high signal loss whereas green indicate areas of low signal loss. The small rise in the middle left of the profile shows an area of increased signal loss directly to its right and behind it relative to the transmitter location.

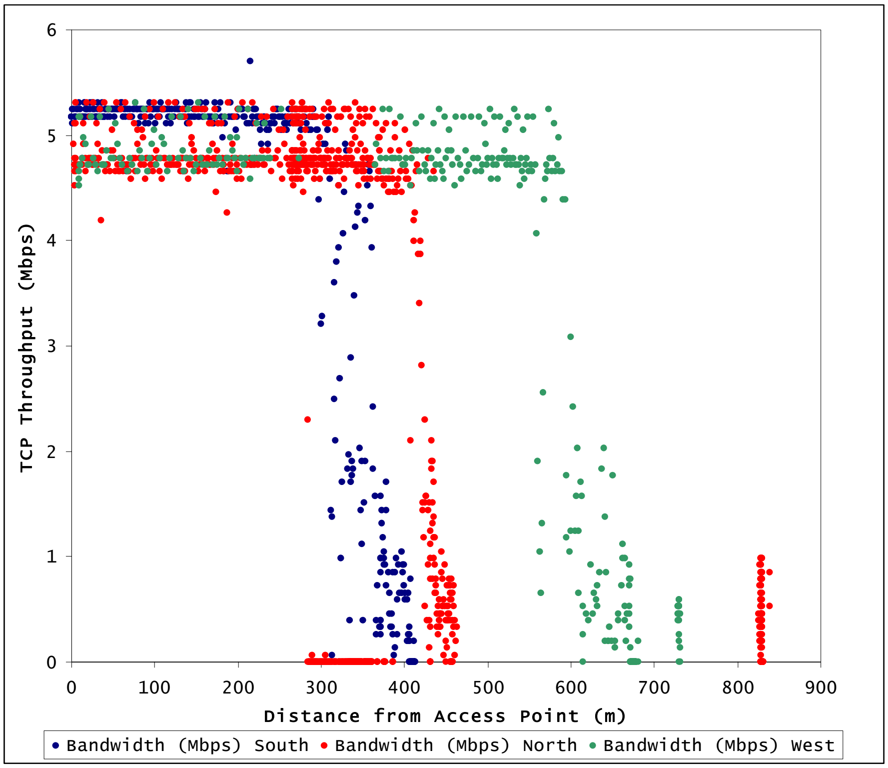

Figure 10.—TCP Throughput over distance at the Dry Valley Test Site near Hanksville, Utah as recorded by GPSIPerf. Approaches to a stationary access point were made from the south, north, and west. Dependency on terrain features and line-of-sight signal propagation may have caused the differences in the distance at which throughput first begins to change. Evidence for rapidly fluctuating adapter rates was also seen at 0.73 and 0.83 km in throughput measurements for western and northern traverse legs, respectively.

3. GPSIPerf Field Tests

GPSIPerf was tested during the first week of May 2004. Two field sites and three different hardware access point configurations were used during this field testing period. Field conditions were dry to very dry, and temperatures were between 85 and 97 °F.

The first of the testing sites was south of Hanksville, Utah in Dry Valley. The purpose of tests performed here was to establish the throughput capabilities of non-elevated and elevated (fig. 11) commodity access points to clients at varying distances over relatively uniform terrain. These tests served as reference points to later testing done with more complex access point configurations.

The second testing site was at Factory Butte. This location provided long flat stretches of road to perform bandwidth throughput distance testing of Tropos repeaters as well as Vivato, Inc., antenna arrays. Furthermore, a dry wash at this site was used to gather bandwidth throughput measurements for clients in environments with complex geometry. The latter form of testing was referred to as “canyon testing.”

3.1.1 Dry Valley Test Site—Ground Level Access Point.—The access point (AP) hardware configuration consisted of an approximately 2 m high, tripod-mounted Linksys WAP11 802.11b wireless access point (30 mW output) situated at 38.294 N. latitude, 110.715 W longitude. An “iperf” server was started on a laptop. This computer was connected with a direct cable connection to the Linksys access point. GPSIPerf was run on a separate laptop that was equipped with a DeLorme USB EarthMate2 GPS unit and an Avaya PCMCIA wireless networking card. Configuration of the “iperf” within GPSIPerf used the iperf server laptop’s static IP as the server address and a message interval of 1 transmission per second. TCP messages were configured to a window-size of 256 kB with an 8 kB buffer length. The following command was created by GPSIPerf to execute the iperf client:

[PATH\]iperf.exe -c XXX.XXX.XXX.XXX -l 8K -w 256K -i 1 -f m -t 10000

where *PATH* was the location of the iperf executable and *XXX.XXX.XXX.XXX* is the IP address of the laptop.

The GPS unit was situated on the dashboard of the test vehicle. An external, 8 dBi co-linear antenna was connected to the Avaya card and was magnet mounted to the exterior roof of the vehicle. The vehicle was driven along roads north, south, and west of the access point at a rate not greater than 10 mph.

GPSIPerf was used to measure the TCP throughput during multiple traverses of roads found in the test site. GPSIPerf performed well for these test. Some difficulties acquiring GPS coordinates were encountered while increasing distance from the access point. However, it is not clear at this time whether the difficulties were related to hardware or software components in the test. TCP throughput measurements without valid GPS coordinates were discarded during the data processing phase.

Testing from each of these three cardinal directions driving towards the access point revealed the effects of distance (i.e., signal attenuation) on TCP throughput (fig. 10). The distance at which TCP throughput drops from values ca. 5 Mbps to those around 1 Mbps differs with each of the tested directions. Access point approaches from the south and north revealed rapid increases in the TCP throughput at 400 and 450 m. Approaches from the west showed similar rapid increases in throughput at 700 m. The jump from low to high throughput, which was an indicator of the hardware configuration's capacity for rate adaptation, occurred rapidly. All approaches reached the upper tiers of throughput performance within 50 m once throughput levels began to increase.

We also tested the effects of multiple (i.e., 2) clients on TCP throughput. When maximum data transfer rate settings are used by multiple clients, the total bandwidth available is shared. That is, if the access point was capable of supporting 11 Mbps, then each of the two clients experienced an equal share of the theoretical maximum 5.5 Mbps transfer rate. However, we observed when one of the clients is forced to adopt a lower transfer rate (e.g., 1.1 Mbps), the other client was forced to assume that rate as well.

One final observation from our testing of a non-elevated access point was made. We noticed three distinct throughput tiers with this hardware configuration. The first two tiers are tightly constrained to throughputs of ~5.2 and ~4.8 Mbps. The last tier is broader ranging from 0 to 1.2 Mbps. This tiered structure may be an artifact of the Avaya card's signal-handling and rate adaptation schemes.

3.1.2 Dry Valley Test Site—Elevated Access Point.—Hardware configuration for the elevated access point (AP) tests was similar to that of the ground level test. The only change was in the positioning of the access point to a higher elevation (fig. 11). The difference in elevation between the ground access point (1338.4 m) and the elevated access point (1390.6 m) was ~50 m.

We observed increased throughput at distances comparable to those sampled during the ground level AP tests (fig. 12). TCP throughput at levels up to 3.8 Mbps was measured at a

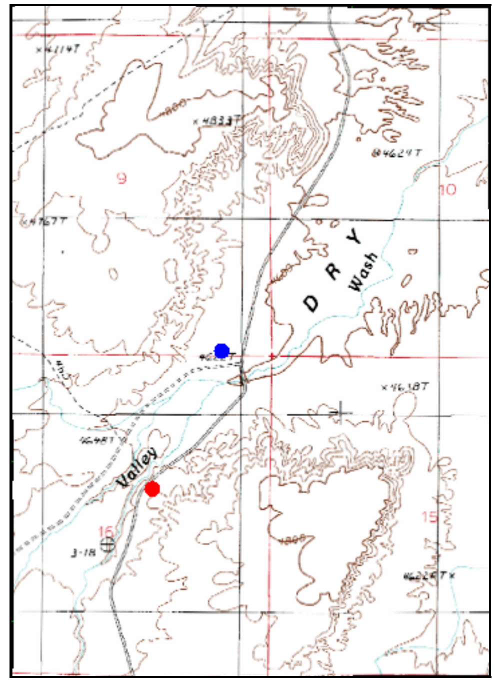

Figure 11.—Wireless Access Point locations used for ground-level and elevated TCP through put testing at the Dry Valley Site. Difference in elevation between ground-level (Blue dot) and elevated (Red dot) AP locations elevation was approximately 50 m.

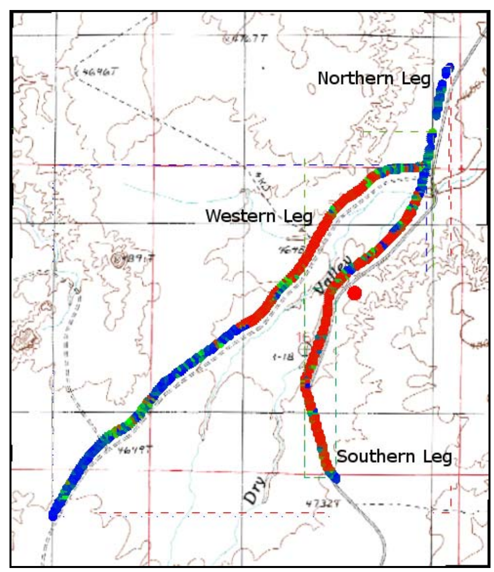

Figure 12.—Elevated access point testing. The affects of elevated access points on wireless TCP throughput were apparent. We measured higher throughputs at greater distances using an elevated access point. The area on the Western Leg shows decreased throughput that may be due to multipathing of the wireless signal. Distances at which hardware adapter speed varies rapidly are also apparent.

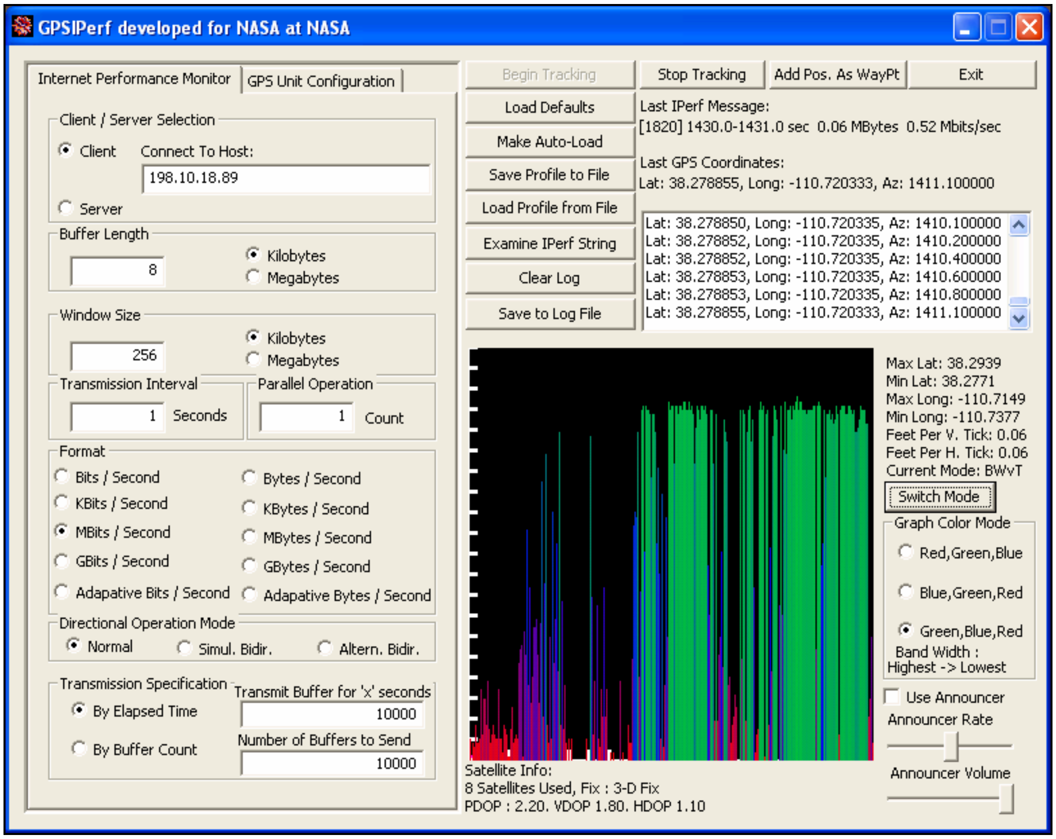

Figure 13.—Rapid Changes in adapter speed were seen in the GPSIPerf TCP throughput data. We observed rapid fluctuations in TCP throughput (red and blue lines) over time along the Western Leg at the Dry Valley test site.

distance of 2 km on the western road. Throughput at this level utilized the Avaya's 5.5 Mbps rate adaptation scheme.

Several observations were made during these tests that were similar to earlier notes made by the research team regarding the distance (~0.8 km) at which wireless driver rate adaptation algorithms have difficulty maintaining a steady connection with the changes in signal from the access point. These rapid changes in adapter speed were apparent in the GPSIPerf throughput data (fig. 13).

3.2.1 Factory Butte—Tropos Mesh Network.—Tropos (ref. 20) wireless mesh network hardware (Wi-Fi Cell Outdoor Unit 5110) with 1W Tx output was used during tests conducted at Factory Butte, Utah. This unit offers broader coverage to 802.11b networks as well as adding rapid deployment capabilities. Tropos units were configured to test their ability to extend wireless coverage with a GPSIPerf client. Hardware configuration was otherwise the same as that used in the Dry Valley tests.

Tropos unit testing revealed that the use of Tropos mesh network units can increase wireless network coverage quite dramatically. TCP throughput remained high (~1 Mbps) even at distances over 4 km from the access point (fig. 14). The pattern of rapid rate changes from 5 Mbps to 2.4 Mbps at (0.8 km) from the access point continued. The distance interval (0.8 to 1.6 km) over which the fluctuations occurred appeared to be extended to a distance of about 800m by the presence of a Tropos node in the vicinity. This extension of the rate fluctuation interval may have been due to the inability of the card to rapidly associate with the more powerful signal of the Tropos node.

A second series of rapid fluctuations from rates of 2.4 Mbps to 1.7 Mbps occurred at a distance of 3.8 km and may be associated with a step down in the card's transfer rates. Additionally, it is possible that the observed transition from 1.7 Mbps rates to 0.9 Mbps at a distance of 4.1 km is another indication of rapid rate fluctuations. We also observed that once the mesh network node was in wireless range, it took some time for the wireless networking card to associate with it.

3.2.2 Factory Butte—Vivato Antenna Array.—One of the more interesting tests conducted at the Factory Butte site was "canyon" test. This test utilized a Vivato VP1200 indoor Wi-Fi panel antenna (ref. 21). This antenna panel was enabled with packet-steering technology. Packet-steering allowed any one of 13 panels along an arc of 104° to become the dominant 802.11b signal source for clients. The source panel could change as clients moved through the array's signal arc. The antenna panel consisted of 13 access points with varying

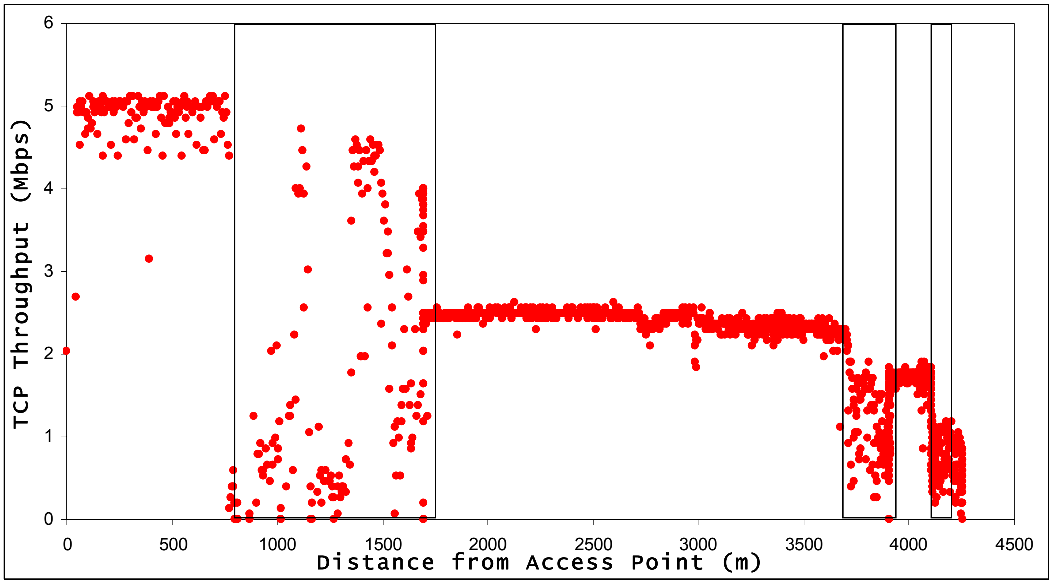

Figure 14.—TCP Throughput over Distance. TCP Throughput measurements remained high to a distance of 4 km using Tropos mesh networking hardware. Distance intervals where rapid fluctuations in TCP throughput are highlighted with blue boxes. These fluctuations may have been caused by slow wireless card rate adaptation algorithms and AP association schemes.

signal strengths. The variation in signal strength depended on the client location relative to individual panels within the Vivato array. An important observation from our testing of this antenna was the clear need for the development of open-source or intelligent fast-switching client network card driver software. We noted on several occasions during testing that our GPSIPerf client switched from one good access point to a weaker access point for reasons we do not understand[1]. We also noted that the time it took our Windows applications stack to handle the handoff from one AP to another was quite long, on the order of 200ms to 2 or 3 seconds. Fast-switching client software that "spoofs" the operating system above it to make the seamless switch between AP's would be useful for future GPSIPerf testing.

Canyon testing was used to simulate the effect of complex geometries on 802.11b signal and TCP throughput when doing ground-based research. Such environments may occur at other sites on Earth, the Moon or Mars. Although the test setup was not intended for scientific study of radio propagation in a desert canyon, the test was designed in a manner that was useful to assess the utility of the GPSIPerf tool. Future experiments could be designed such that the GPSIPerf tool is used to map TCP throughput over the smaller fractal dimensions (i.e., nooks and crannies) represented within the canyon using differential GPS technology/equipment.

A dry wash running perpendicular to the road used for the Tropos testing was used for this testing. The Vivato antenna array panel was placed about 10 m behind the lip of a mound overlooking the canyon, approximately 20 m above the dry wash bed.

No vehicle was used in these tests. We walked the Coal Mine Wash carrying a laptop that was equipped with a Avaya 802.11b card and a DeLorme Earthmate 2 GPS unit. Carrying the laptop had an attendant possibility of shielding the wireless card's antenna from the Tropos array with the laptop carrier's body. Therefore, appropriate precautions were made to avoid standing directly in the line-of-sight path.

The results indicated that, in general, the complex geometries of the wash caused 802.11b coverage and thus TCP throughput measurements to be variable over small geographic distances (fig. 15). However, a plateau of 4.9 Mbps was seen at a distance of 350 m from the access point. These levels were not consistent even within the first 100 m. TCP throughput dropped somewhat at distances over 350 m revealing rapid fluctuations in adapter speed between 5.0 Mbps and 1.0 Mbps rates, and failed entirely before the 500 meter mark was reached.

As expected, large obstacles to the line-of-sight between laptop and access point, such as boulders and other natural rock formations, caused interruptions in TCP throughput. The exact positions of these features relative to the client and the access point were not noted so only this qualitative correlation was noted.

[1]This is one of the problems with working with commercial off the shelf technology with closed-source driver software.

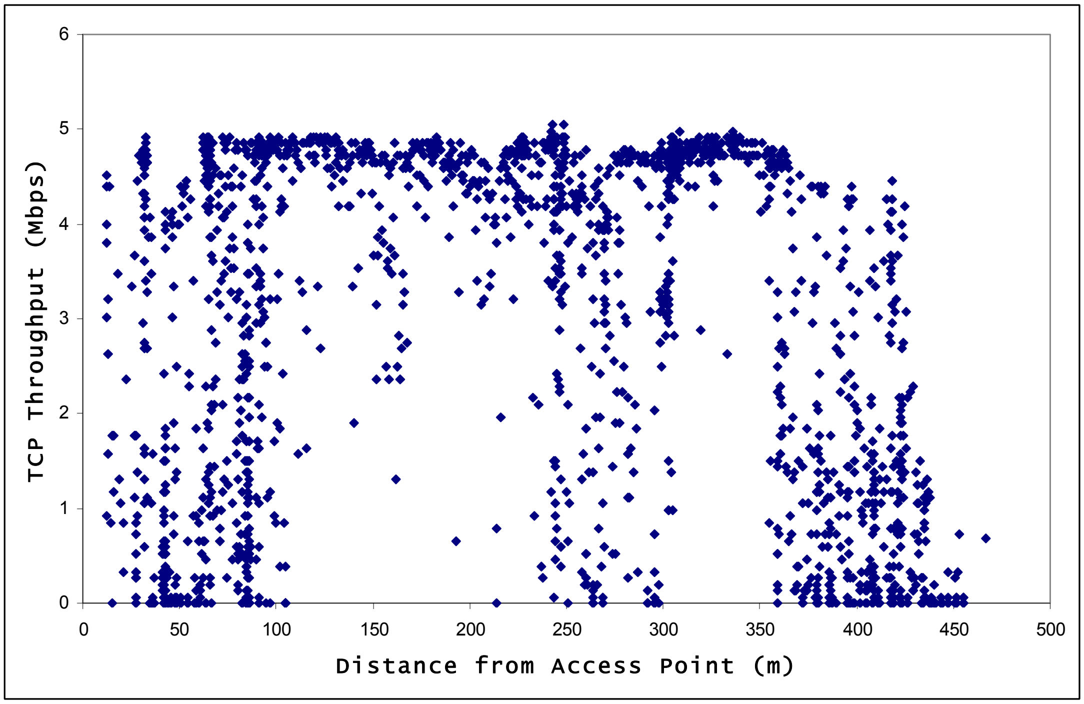

Figure 15.—Canyon Testing with the Vivato Antenna Array. Multiple Areas of very low TCP Throughput were found during our testing of GPSIPerf at the Factory Butte site's Coal Mine Wash. These areas of low bandwidth included regions within 100 meters of the Access Point. 802.11b coverage was spotty but provided relatively high levels of throughput up to ~360 meters.

5. Discussion

Use of GPSIPerf for field testing wireless hardware configurations demonstrated that TCP throughput was highly susceptible to hardware driver rate-adaptation, distance, access point configuration and the geometry of the terrain over which the signal is to propagate. These observations were consistent with both RF propagation theory and the observation of other researchers where multipath, attenuation of signal, and signal strength are all factors in determining the response of a wireless network.

GPSIPerf provided a clear picture of TCP throughput conditions in our remote wireless network sites. One application of GPSIPerf that immediately suggests itself is the characterization of application level network performance in more conventional, terrestrial wireless network architectures. However, there also exists the possibility of the use of GPSIPerf and similar tools in extraterrestrial environs.

GPSIPerf may be utilized to gain insight into *n*-tier application network performance and data transfer between automated and/or remote wireless network computing platforms. Furthermore, GPSIPerf data might also be used to make informed, and (potentially) automated, decisions about when and where such transactions can best take place. This is important for the next generation of space exploration, because it seems that propagation of data *via* wireless networks will be the *de facto* approach to performing remote science. Current simulations and experiments initiated with the intent of providing a baseline for interplanetary exploration activities are already taking this approach. Platforms such as the Lunar and Planetary Science Module (ref. 21) are using wireless data transfer to effectively extend NASA's capability to perform remote science.

Visualization tools such as GPSIPerfView can also be used as an aid to gain insight into observed TCP throughput performance as well as to detect and plan around areas of potentially poor performance. GPSIPerf supplies accurate TCP throughput measurements that may be geographically referenced within GPSIPerfView to give the operators of these platforms a sense of terrain during missions. Furthermore, GPSIPerfView offers operators the ability to plan data transfers according to surrounding terrain and prevailing RF conditions. For example, the effect of terrain features such as hills and valleys on RF propagation can be clearly seen using GPSIPerfView's Irregular Terrain Model Visualization tool (fig. 16). Areas with increased RF loss can be accurately identified. Two approaches to exploration suggest themselves:

1) These areas can either be avoided when continuous communications are desired.
2) These areas can be visited and data collected can be transferred when the remote platform has moved to environs that are more conducive to data transfer.

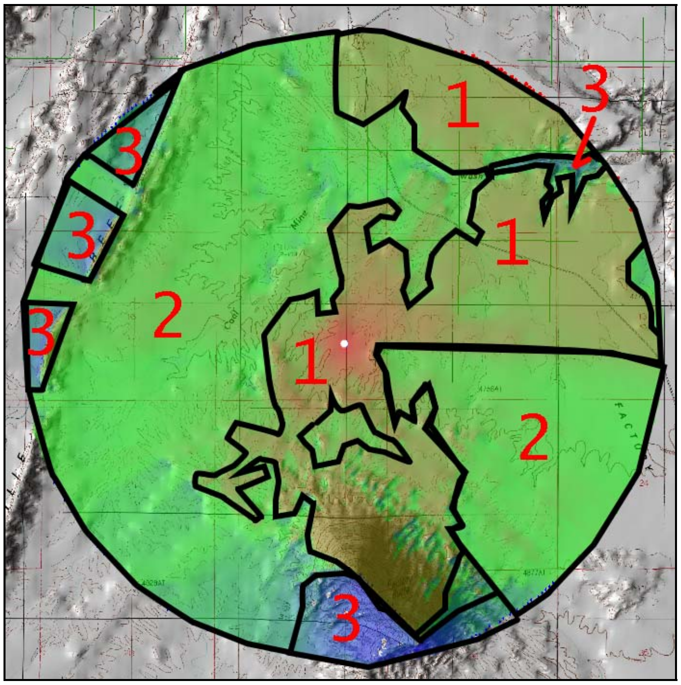

Figure 16.—GPSIPerf View RF Solution for Factory Butte, Utah. GPSIPerfView is capable of providing a geographical context for RF propagation. Test scenarios can be planned according to modeled RF losses. Testing runs in which data must be transferred continuously from clients via wireless networks may be restricted to areas marked with ‘1’. Autonomous exploration of areas marked with a ‘3’ would require subsequent data transfer from locations with ‘1’s or ‘2’s.

The first approach might be appropriate for selecting and sampling along a TCP throughput-optimized route for reconnaissance to a known destination. The second approach is useful for exploration of areas similar to those found during our canyon tests at the Factory Butte where TCP throughput is erratic and not consistent with projected RF loss predictions.

While GPSIPerf and GPSIPerfView provide users with useful information, there is room for improvement. For example, GPSIPerf could provide users with more information regarding the physical parameters of the network connection. Such parameters include but are not limited to signal quality, signal strength and noise. These parameters may not be of interest to those who are primarily interested in application level network performance, but they can often be informative to researchers who are concerned with more esoteric aspects of observed wireless networks.

GPSIPerfView also has room for improvement, particularly in its use of the Irregular Terrain Model. The model, as it is implemented, is limited to a distance range between 1 and 20 km. Network hardware such as the Tropos Mesh Network access points has an effective range well in to the 5 km range from what we have observed. However, common wireless network hardware such as the Linksys access points used at the Dry Valley test have lower power outputs (30 mW) and are limited to ranges of less than 1 km. It is possible that lower power requirements could force networking hardware intended for extraterrestrial exploration to have similar low power outputs. Thus, at some point, it may become necessary to provide RF propagation models that are accurate over distances less than 1 km. Similarly, efforts should be made to provide updated models that are capable of modeling and accounting for RF multipathing and Fresnel zones. The addition of these parameters and considerations would invariably allow applications such as GPSIPerfView to provide users with more accurate depictions of the RF environment.

References

1. Netstumbler. URL: http://www.netstumbler.com.
2. Kismet. URL: http://www.kismetwireless.net/index.shtml.
3. IPERF Software. URL: http://dast.nlanr.net/Projects/Iperf.
4. United States Geological Survey. URL: http://www.usgs.gov.
5. Irregular Terrain Model Implementation. URL: http://elbert.its.bldrdoc.gov/itm.html.
6. DeLorme Earthmate 2 URL: http://www.delorme.com/earthmate.
7. Sager Notebook URL: http://pctorque.com.
8. DeLorme Drivers URL: http://www.delorme.com/support/gps/uemate005.asp.
9. Object Management Group. URL: http://www.omg.org.
10. MICO CORBA ORB. URL: http://www.mico.org.
11. *Firewall Traversal for CORBA Applications Using an Implementation of Bidirectional IIOP in MICO.* Robert Griffin. NASA/CR—2002-211979. October 2002.
12. SDTS Data. URL: http://mcmcweb.er.usgs.gov/sdts.
13. SDTS++ Source Code. URL: http://mcmcweb.er.usgs.gov/sdts/sdtsxx/index.html.
14. BOOST Libraries. URL: http://www.boost.org.
15. FIPS123 Libraries. URL: http://thor-f5.er.usgs.gov/sdts/software/fips123.
16. NASA Glenn Research Center GRUVE. URL: http://facilities.grc.nasa.gov/gruve/gruve_quick.html.
17. OpenGL View Code. URL: http://www.falappa.net/alessandro/programming/glenabledview.html.
18. KDEM Code. URL: http://www.mindspring.com/~jamoyers/kdem.
19. Irregular Terrain Model Authors and C++ Dynamic Link Library. URL: http://elbert.its.bldrdoc.gov/itm.html.
20. Tropos Hardware. URL: http://www.tropos.com/pdf/5110_datasheet.pdf.
21. R.A. Beck, R.K. Vincent, D.R. Watts, M. Seibert, D. Pleva, M. Cauley, C. Ramos, T. Scott, D. Harter, J. Kosmo, A. Ross, K. Groneman, J. Rojas, NASA mobile lunar and planetary science module, in: 34th Annual Lunar and Planetary Science Conference, March 2003.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 2005	Final Contractor Report

4. TITLE AND SUBTITLE

New Applications for the Testing and Visualization of Wireless Networks

5. FUNDING NUMBERS

WBS–22–302–15–40
1L161102AF20

6. AUTHOR(S)

Robert I. Griffin, Michael A. Cauley, David P. Pleva, Marc A. Seibert, and Isaac Lopez

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
John H. Glenn Research Center at Lewis Field
Cleveland, Ohio 44135–3191

8. PERFORMING ORGANIZATION REPORT NUMBER

E–15047

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Washington, DC 20546–0001
and
U.S. Army Research Laboratory
Adelphi, Maryland 20783–1145

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA TM—2005-213580
ARL–MR–610

11. SUPPLEMENTARY NOTES

Robert I. Griffin, RS Information Systems, Inc., 21000 Brookpark Road, Cleveland, Ohio 44135; Michael A. Cauley and Marc. A. Seibert, NASA Glenn Research Center; David P. Pleva, Verizon Federal Network Systems, Arlington, Virginia 22209; and Isaac Lopez, U.S. Army Research Laboratory, NASA Glenn Research Center. Responsible person, Robert I. Griffin, organization code PRV, 216–433–2382.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified - Unlimited
Subject Category: 61

Available electronically at http://gltrs.grc.nasa.gov

This publication is available from the NASA Center for AeroSpace Information, 301–621–0390.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Traditional techniques for examining wireless networks use physical link characteristics such as Signal-to-Noise (SNR) ratios to assess the performance of wireless networks. Such measurements may not be reliable indicators of available bandwidth. This work describes two new software applications developed at NASA Glenn Research Center for the investigation of wireless networks. GPSIPerf combines measurements of Transmission Control Protocol (TCP) throughput with Global Positioning System (GPS) coordinates to give users a map of wireless bandwidth for outdoor environments where a wireless infrastructure has been deployed. GPSIPerfView combines the data provided by GPSIPerf with high-resolution digital elevation maps (DEM) to help users visualize and assess the impact of elevation features on wireless networks in a given sample area. These applications were used to examine TCP throughput in several wireless network configurations at desert field sites near Hanksville, Utah during May of 2004. Use of GPSIPerf and GPSIPerfView provides a geographically referenced picture of the extent and deterioration of TCP throughput in tested wireless network configurations. GPSIPerf results from field-testing in Utah suggest that it can be useful in assessing other wireless network architectures, and may be useful to future human-robotic exploration missions.

14. SUBJECT TERMS

GPS; Wireless communication; Applications; DEM; ITM

15. NUMBER OF PAGES

22

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	